

HOSTED BY



Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

journal homepage: www.elsevier.com/locate/jestech

Full Length Article

Energy and area efficient hierarchy multiplier architecture based on Vedic mathematics and GDI logic

Mohan Shoba*, Rangaswamy Nakkeeran

Department of Electronics Engineering, School of Engineering and Technology, Pondicherry University, Puducherry 605014, India

ARTICLE INFO

Article history:

Received 19 April 2016

Revised 10 June 2016

Accepted 19 June 2016

Available online xxxx

Keywords:

Multiplier

FS-GDI logic

CslA

BEC converter

4-2 compressor

ABSTRACT

Hierarchy multiplier is attractive because of its ability to carry the multiplication operation within one clock cycle. The existing hierarchical multipliers occupy more area and also results in more delay. Therefore, in this paper, a method to reduce the computation delay of hierarchy multiplier by employing CslA and Binary to Excess 1 Converter (BEC) is proposed. The use of BEC eliminates the $n/4$ number of adders, existing in the conventional addition scheme, where n denotes the multiplier input width. As the area of the hierarchy multiplier is determined by its base multiplier, the base multiplier is realized with the proposed Vedic multiplier, which has small area and operates with less delay than the conventional multipliers. In addition, the reduction of power consumption in the hierarchy multiplier can be ensured by implementing the designed multiplier with full swing Gate Diffusion Input (GDI) logic. The performances of the proposed and the existing multipliers are evaluated by Cadence SPICE simulator using 45 nm technology model. From the simulation results, the performance parameters namely, delay and power consumption are calculated. Further, the area is measured from the corresponding layout for the same technology model. It is examined from the results that the proposed multiplier operates with 17% lesser power delay product than the recently reported hierarchy multiplier. The Monte Carlo simulation is performed to understand the robustness of the proposed hierarchy multiplier.

© 2016 The Authors. Production and hosting by Elsevier B.V. on behalf of Karabuk University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

Hierarchical multipliers are considered as viable means for achieving orders of magnitude speed up in computer intensive applications through the use of fine grained parallelism. They are used in various fields of numerical and scientific computations, image processing, communication, cryptographic computation and so on [1–5].

In general to design n bit hierarchical multiplier, four $n/2$ base multipliers are necessary which generate $2n$ bit output, where n represents hierarchical multiplier input width. It is noted that all the base multipliers are allowed to perform the task in parallel. Due to that, the performance of the hierarchy multiplier is determined from the accumulation delay of its base multipliers output bits. But this is a time consuming task as it requires more number

of additions and considered a bottleneck for the hierarchy multiplier performance. In this work, an approach to perform this accumulation with less number of addition process is proposed. The following are the contributions discussed in the paper:

- (i) For the area and delay efficient implementation of base multiplier, a new design is proposed based on Vedic mathematics concept.
- (ii) To reduce the accumulation delay of base multiplier output bits, CslA (CslA) and Binary to Excess 1 Converter (BEC) are introduced.
- (iii) To realize the hierarchy multiplier with small area it is realized using Full Swing Gate Diffusion Input (FS-GDI) logic.

The rest of the paper is organized as follows: An overview of the hierarchy multiplier is described in Section 2. In Section 3, the explanation of the proposed hierarchy multiplier and the implementation of its building components namely, base multiplier, CslA adder, BEC converter are also given. The simulation results and discussion are given in Section 4 and finally, the Section 5 concludes the paper.

* Corresponding author.

E-mail addresses: shobamalar@gmail.com (M. Shoba), nakkeeranpu@gmail.com (R. Nakkeeran).

Peer review under responsibility of Karabuk University.

<http://dx.doi.org/10.1016/j.jestech.2016.06.007>

2215-0986/© 2016 The Authors. Production and hosting by Elsevier B.V. on behalf of Karabuk University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

2. Hierarchy multiplier

Multipliers with large width are required for the implementation of cryptography and error correction circuits for more reliable transmission over highly insecure and/or noisy channels in networking and multimedia applications. The hierarchical principle helps to realize fast large bit multiplier, except that it requires a large width adder for performing the addition task, which poses limitation on the performance and increases area of the designed multiplier [6–7].

Over the last few decades, a lot of works have been dedicated, at the algorithmic and implementation level, to improve the performance of hierarchical multiplier. The delay in the addition process of the hierarchy multiplier is reduced with the parallel execution of ripple carry adder [8]. However, this method requires twice the number of adders thus results in increased area. On the other hand, the delay is reduced with the deployment of carry look ahead adder for the addition process but this increases the interconnection complexity [9]. Not only delay and area, the power consumption of the hierarchy multiplier also has to be reduced because the existing designs appending more zeros to equalize the number of bits in order to make them suitable for parallel computation [10]. This might increase the spurious activities and thus increases the power consumption. The above mentioned issues in the existing hierarchy multiplier can be addressed by

- (i) Incorporating BEC to eliminate $n/4$ number of adders at the final stage of addition process.
- (ii) Performing the final addition using proposed CslA.
- (iii) Implementing the proposed hierarchy multiplier using FS-GDI logic as carried out in this paper.

3. Methodology

In this section, an approach for efficient implementation of n bit hierarchy multiplier with minimum delay is discussed. As an example, the architecture for 16 bit multiplier design is explained. Further, a new design is suggested for the hierarchy multiplier building block namely, base multiplier based on Vedic mathematics. Following that, the discussion of CslA, binary to excess 1 converter and GDI logic is carried out in this section.

3.1. Proposed hierarchy multiplier

In general, the hierarchy multiplier speed is determined from the computation delay of base multiplier output bits addition. This delay can be decreased by minimizing the number of additions without affecting the functionality. The following approach is incorporated in the proposed n bit hierarchy multiplier multiplication procedure to reduce the delay:

- Step 1: The multiplier inputs and output are represented as X , Y and Z , respectively.
- Step 2: Divide n bit multiplier inputs i.e., X and Y , into equal two halves. For the input X , it is divided into $(X_{n/2-1}, \dots, X_0)$, $(X_n, \dots, X_{n/2})$, which are assigned as X_L and X_H , respectively. The same procedure is also adopted for another multiplier input Y .
- Step 3: After dividing both the inputs, they are formed into four groups like (X_L, Y_L) , (X_H, Y_L) , (X_L, Y_H) and (X_H, Y_H) .
- Step 4: The multiplication is accomplished using four $n/2$ bit base multipliers namely, a_0 , a_1 , a_2 and a_3 .
- Step 5: The multiplier product bits $Z_{n/4-1}, \dots, Z_0$ is obtained from 0 to $n/2-1$ output bits of a_0 .

- Step 6: The resultant bits of a_1 , a_2 and concatenation of a_0 ($n/2$ to n), a_3 (0 to $n/2-1$) are formed an array of carry save format which are processed by carry save adder.
- Step 7: The resultant sum and carry from carry save adder are becoming the inputs for CslA of n bit adder. Also, the sum output of CslA adder are assigned as multiplier resultant bits for the range of $Z_{n+n/2-1}, \dots, Z_{n/2}$.
- Step 8: BEC takes the input from a_3 ($n/2$ to n bit) and its output bits are available prior to CslA and they are passed to the multiplexer.
- Step 9: The multiplier output bits $Z_{2n} \dots Z_{n+n/2}$ is obtained from the multiplexer, based on the carry output of CslA adder, if it is one then the BEC output are becoming the output otherwise the product bits of a_3 ($n/2$ to n bit).

Based on this algorithm, 16 bit (n) hierarchy multiplier architecture is designed as shown in Fig. 1. The multiplier inputs are X , Y of 16 bit width and produces the output Z of 32 bit. First, the inputs X and Y are divided into equal two halves namely, X_H and X_L , Y_H and Y_L and they are multiplied by 8 bit base multiplier. As seen in Fig. 1, the symbols of a_0 , a_1 , a_2 and a_3 denote the base multiplier for the multiplication of $(X_L$ and $Y_L)$, $(X_H$ and $Y_L)$, $(X_L$ and $Y_H)$ and $(X_H$ and $Y_H)$, respectively. Once these multiplication processes is over, then their output bits will form a carry save array as per step 6, which in turn processed by carry save adder thus results into two rows of 16 bit output. These bits are further added with the help of 16 bit CslA adder to produce the Z_{24}, \dots, Z_8 multiplier output bits. Meanwhile, the BEC also computed its output and fed to multiplexer as one of the inputs. Another input for the multiplexer is from a_3 output (half of the output bits i.e., $n/2$ to $n-1$). Finally, the multiplexer selects, either BEC or a_3 output bit as $Z_{24}-Z_{31}$, based on CslA adder's carry.

As a result of introduction of BEC in the hierarchy multiplier, $n/4$ adders are eliminated. Due to the parallel computation of BEC and CslA output, the processing delay for multiplier output bits i.e., $Z_{24}-Z_{31}$ is minimized significantly. As seen from the architecture of proposed hierarchy multiplier, given in Fig. 1, the critical path of the proposed architecture consists of one base multiplier, one bit adder, one CslA adder and multiplexer only. Further, the implementation details of building components of the hierarchy multiplier namely, base multiplier, CslA adder and BEC converter are described in the following subsection.

3.2. Base multiplier

As discussed in the earlier section, the performance of the hierarchy multiplier is determined by its base multiplier. In the conventional multiplication techniques, the intermediate computation involved in the multiplier operation reduces the speed exponentially in accordance with the number of bits present in multiplier input. This becomes critical issue for more number of input bits. But this issue can be mitigated by the parallel addition of partial products which is an inherited principle of Vedic multiplication method. Though partial products reduction is possible in Booth multiplication, the encoding and decoding mechanism involved in this method increases the circuit complexity thereby power consumption. On the other hand, Wallace multiplication uses random placement of counters for the efficient partial product accumulation thus makes the design becomes complex than the conventional scheme. Therefore, the Vedic multiplication is considered as an alternative way of performing the multiplication operation without increasing the circuit complexity and power consumption [11–14]. In this multiplication process, the partial products are accumulated at every step as opposed to the conventional multiplication schemes. Therefore, the speed of this

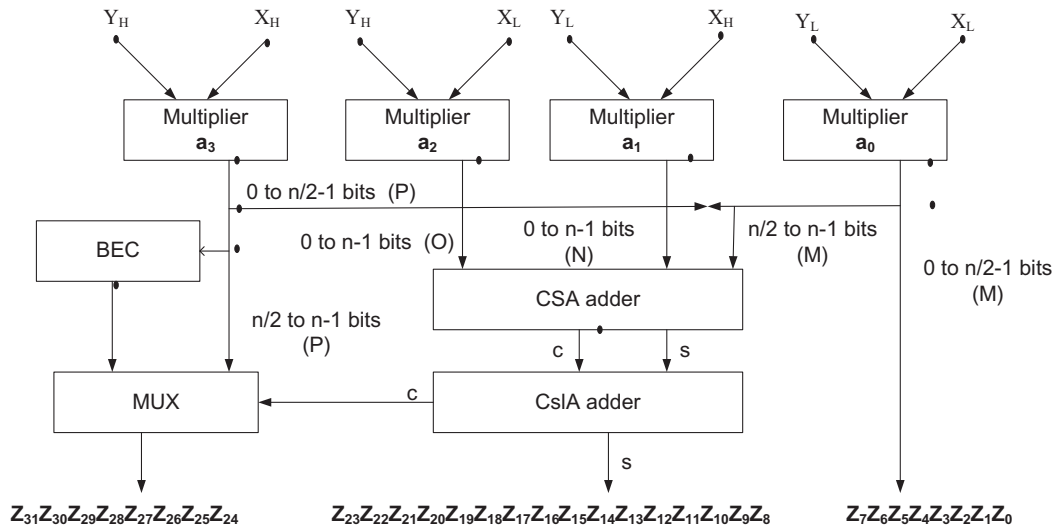


Fig. 1. Proposed 16 bit hierarchy multiplier.

multiplier can be improved by reducing its partial product accumulation delay. This is attempted in the proposed 8 bit multiplier and its representation is shown in Fig. 2.

The multiplier inputs and outputs are represented as X_i , Y_i and P_{2i} , where i is 0 to $n - 1$, n denotes the input bit width (for 8 bit multiplier, $n = 8$). The multiplier partial products ($X.Y$) are generated using *AND* gates. From them, the partial product of $X_0.Y_0$ is output bit of multiplier, i.e., P_0 , whereas the remaining output bits are obtained after two stage computation. In the first stage, the partial products generated from *AND* gates which are added using adder. After each addition process, sum and carry are computed and they move into second stage. It is noted that carry free addition is performed in this stage. Also, these output bits including sum and carry from the first stage are not exceeding more than five bits. Therefore, 4-2 compressor is considered for addition of these bits rather than full adder, which is used in the existing scheme. Due to the use of 4-2 compressor, carry free addition is ensured in the second stage too.

3.3. 4-2 Compressor

The existing 4-2 compressors designs exhibit hardware redundancy due to the requirement of separate circuits for sum and carry outputs computation [17–21]. This problem can be addressed by simplifying compressor output Boolean expression without affecting its functionality. The carry output (c) of compressor is same as carry input, if *XOR* output of c_i and x_4 (N) is low, otherwise, it follows the $(x_1 \oplus x_2 \oplus x_3)$ (M) output, where x_1, x_2, x_3, x_4 and c_i are 4-2 compressor inputs. The sum output (s) is obtained by *XOR* operation all these inputs. Similarly, the horizontal carry (c_0) is obtained by multiplexing the input x_1/x_3 depends on the *XOR* output of x_1 and x_2 . The output expression of 4-2 compressor is given in the following Eqs. (1)–(5).

$$M = x_1 \oplus x_2 \oplus x_3 \quad (1)$$

$$N = c_i \oplus x_4 \quad (2)$$

$$s = M \oplus N \quad (3)$$

$$c_0 = (x_1 \oplus x_2)x_3 + \overline{(x_1 \oplus x_2)}x_1 \quad (4)$$

$$c = MN + \bar{N}c_i \quad (5)$$

The architecture of the proposed 4-2 compressor is shown in Fig. 3.

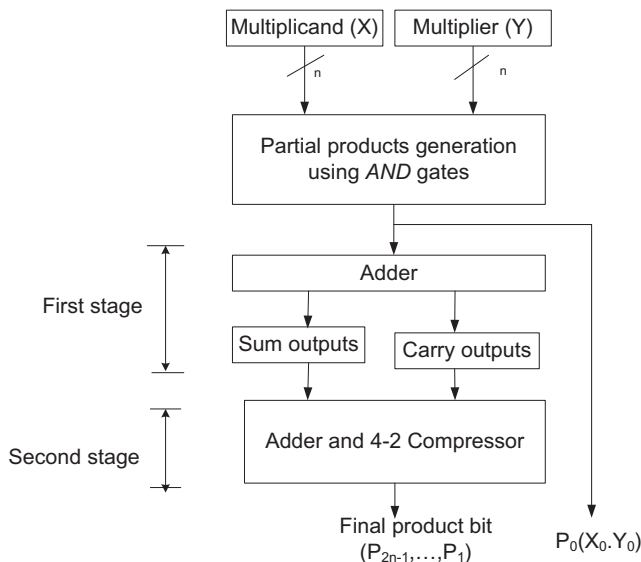


Fig. 2. Block diagrammatic representation of proposed Vedic multiplier.

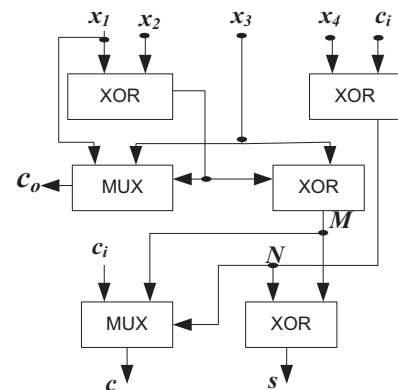


Fig. 3. Proposed 4-2 compressor architecture.

3.4. Carry select adder

There are various adders employed for the addition of base multiplier product bits. They are namely, ripple carry, carry look ahead, carry select and prefix adder. It is well known from the performance study of these adders that CslA has modest performance in terms of area and delay [22–24]. The existing architectures for CslA implementation either suffered by more area (conventional CslA) or by delay (modified CslA). In the modified CslA, the carry computation part uses part of the half adder output as input thereby delay is getting increased. This issue can be overcome by making independent carry computation. Though the gate counts are increasing due to the requirement of separate circuits for carry output, the proposed CslA total layout area will be small due to FS-GDI logic based implementation.

3.5. Binary to excess 1 converter

To reduce the delay of partial products addition in the hierarchy multiplier, this work uses BEC instead of adder for the output bits of $Z_{2n-1} \dots Z_{2n+n/2}$. For n bit input width, $n+1$ bit BECs are required. A structure of 4 bit BEC is shown in Fig. 4.

3.6. GDI logic

An implementation of hierarchy multiplier with reduced transistor count which possesses sufficient driving capability is done with the help of FS-GDI logic. It is a low power design technique which facilitates the implementation of any logic function with fewer numbers of transistors. FS-GDI logic based gates and adder are discussed in the literature [26–29]. Among them, the designs discussed by Shoba et al. in [29] possess better performance in terms of delay, power consumption and area. Therefore, they can be used in the realization of building components of the proposed hierarchy multiplier namely, AND gate, full adder and 4–2 compressor, CslA adder and binary to excess converter.

4. Simulation results and discussion

In this section, the simulation results of the 16 bit hierarchy multiplier and its basic modules namely, base multiplier, 4–2 compressor, 16 bit CslA and 8 bit binary to excess 1 converter are presented. The performance parameters such as area, delay, power consumption and Power Delay Product (PDP) of the simulated designs are evaluated through the SPICE simulation at 45 nm technology with a supply voltage (V_{DD}) of 1.1 V. Typical transistor sizes, i.e., $(W/L)_p = 240 \text{ nm}/45 \text{ nm}$ and $(W/L)_n = 120 \text{ nm}/45 \text{ nm}$ are considered.

The delay and power consumption are calculated as follows: The delay is computed by accounting the time from the 50% of the input voltage swing to 50% of the output voltage swing for each transition. The maximum delay is treated as worst case delay.

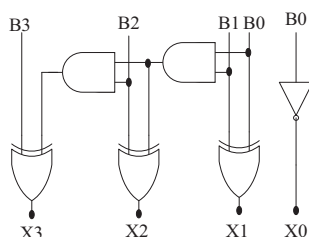


Fig. 4. 4 bit BEC converter circuit [25].

Likewise, the power consumption is determined from the various switching activities and the capacitances of circuit. These procedures are extended for the delay and power consumption calculation of all the simulated modules namely, proposed hierarchy multiplier, base multiplier, 4–2 compressor, CslA and binary to excess 1 converter.

4.1. Proposed hierarchy multiplier

The simulation results of the proposed and existing multipliers are given in Table 1.

4.1.1. Delay

The delay computed through simulation, for all the structures, are given in Table 1 and observed that the proposed multiplier has smaller delay compared to other existing implementations. Due to the deployment of BEC converter in the base multiplier output bits accumulation the number of adders are reduced thus decreases the delay significantly. Moreover, the time taken for the binary to excess 1 converter is not accounted in the critical path delay thereby the speed is improved. The speed improvement obtained by the proposed design is 27% and 11% more than that of multiplier discussed in [2] and [7], respectively.

4.1.2. Power consumption

The power consumed by the simulated hierarchy multipliers is presented in Table 1. The minimum power consumption is witnessed in the proposed design due to the elimination of redundant hardware exhibited in the existing designs thus minimizes the spurious activities. The proposed design has 30% less power consumption than that of existing multiplier.

4.1.3. PDP

The power delay product of the all simulated designs is given in Table 1. Among the multipliers discussed, the best and the worst PDP witnessed corresponds to the proposed and the conventional design, respectively. Also, the energy conservation accomplished with proposed design is 17% more than the multiplier reported in [7].

4.1.4. Area

The area is computed from the layout of simulated multipliers and it is given in Table 1 whereas the layout of the proposed multiplier is given in Fig. 5. From the obtained results, it is witnessed that proposed multiplier has less area. As stated earlier, the FS-GDI is used to implement the basic components of hierarchical multiplier namely, base multiplier, CSA adder, CslA adder, BEC converter with reduced transistor count. Therefore, the area of the proposed hierarchical multiplier is small. The percentage of

Table 1
Simulation results of 16 bit multiplier.

S. no.	Design	Delay (ps)	Power consumption (μW)	PDP (e-15 J)	Area (μm^2)	Frequency (GHz)
1	Multiplier [2]	727	658	478	14,510	1.37
2	Wallace multiplier [30]	657	563	369	14,978	1.43
3	Hierarchy multiplier [7]	594	608	361	15,210	1.68
4	Proposed hierarchy multiplier	528	424	300	12,420	1.89

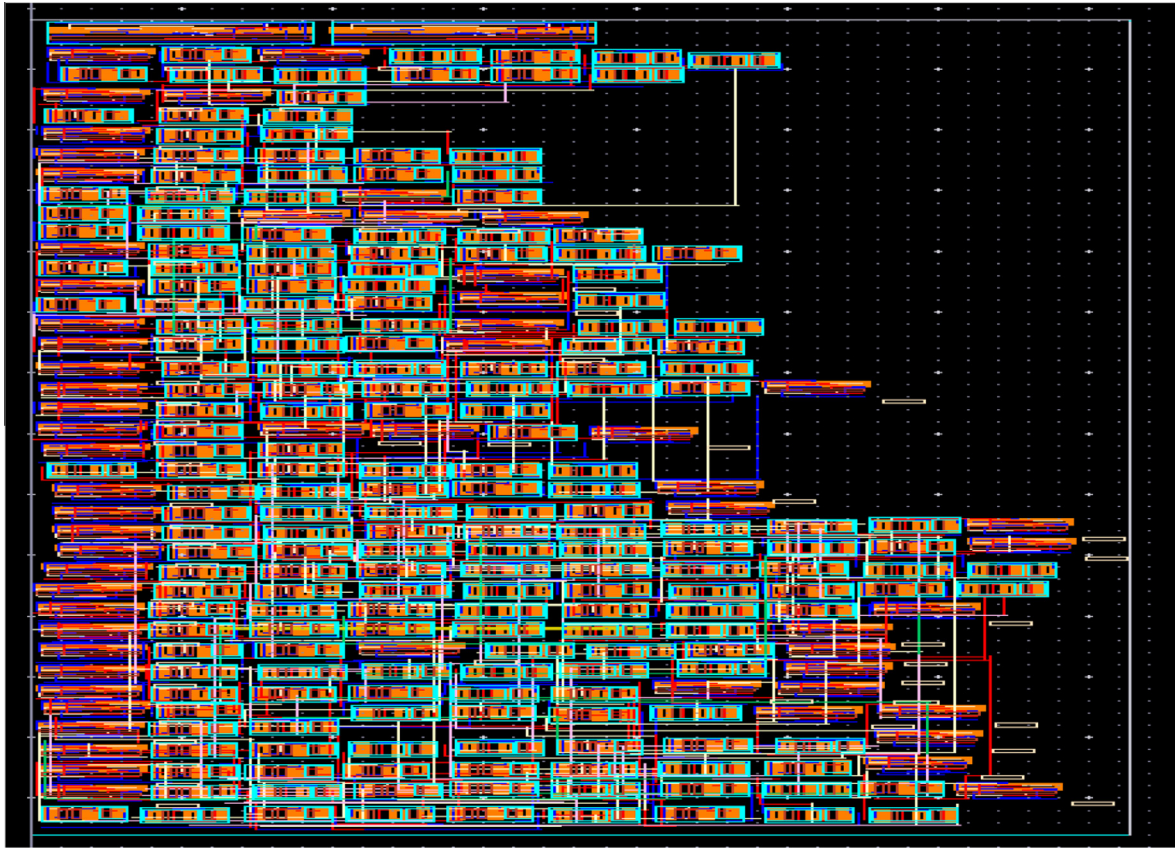


Fig. 5. Layout of the proposed 16 bit hierarchy multiplier.

area reduction possible with proposed design is about 18% more than that of a recently reported multiplier in [7].

4.1.5. Sensitive to process variation

The sensitivity of the circuit's performances namely, delay and power consumption under process variations are studied through Monte Carlo simulations with the thirty runs ($N = 30$). The power consumption distribution results of all the simulated multiplier is given in Fig. 6. It is examined that the proposed hierarchy multiplier has 3% performance deviation under the process changes also.

4.1.6. Voltage variation analysis

The study of circuit's reliability at low supply voltage is recently gained an importance due to increasing demand of battery operated applications such as mobiles, laptops, etc. Therefore, the multipliers circuits are simulated with supply voltage range of 0.7–1.5 V and their power consumption and delay results are tabulated in Tables 2 and 3, respectively. From the obtained results, it is observed that the proposed design has low power consumption than other designs at low supply voltage of 0.7 V.

4.1.7. Load capacitance analysis

To analyze the driving capability of the hierarchy multiplier, all the designs are simulated for a range of capacitance from 2fF to 32fF at nominal temperature of 25 °C with supply voltage of 1.1 V. The simulation results of power consumption and delay are given in Tables 4 and 5, respectively. It can be seen that the performance of hierarchy multiplier based on FS-GDI logic consumes low power even at higher loads than others.

4.2. Base multiplier

The simulation results of base multipliers in respect of delay, power consumption, area and PDP are listed in Table 6.

4.2.1. Delay

The computed delay of the simulated multipliers is given in Table 6. In the proposed multiplier carry propagation is eliminated during the partial products addition which in turn reduces the delay significantly. The speed improvement obtained by the proposed multiplier is 22% more than the multiplier discussed in [16].

4.2.2. Power consumption

The power consumed by the multipliers is computed through simulation and given in Table 6. It is observed from the results that the proposed multiplier design has lesser power consumption than that of existing designs. This is due to the implementation of its building components namely, AND gate, full adder and 4-2 compressor using FS-GDI logic, which minimizes the multiplier transistor count considerably, thereby spurious transitions, thus reduces the overall power consumption. The power saving accomplished in the proposed design is 13% more compared with the Vedic multiplier discussed in [15].

4.2.3. PDP

The power delay product of the proposed and existing multiplier designs is given in Table 6. The power consumption is minimized considerably by implementing the proposed multiplier in FS-GDI logic. Also, the delay is reduced in the proposed multiplier. Thus, the proposed design is conserving 35% more energy (or power delay product) than the conventional multiplier discussed in [8].

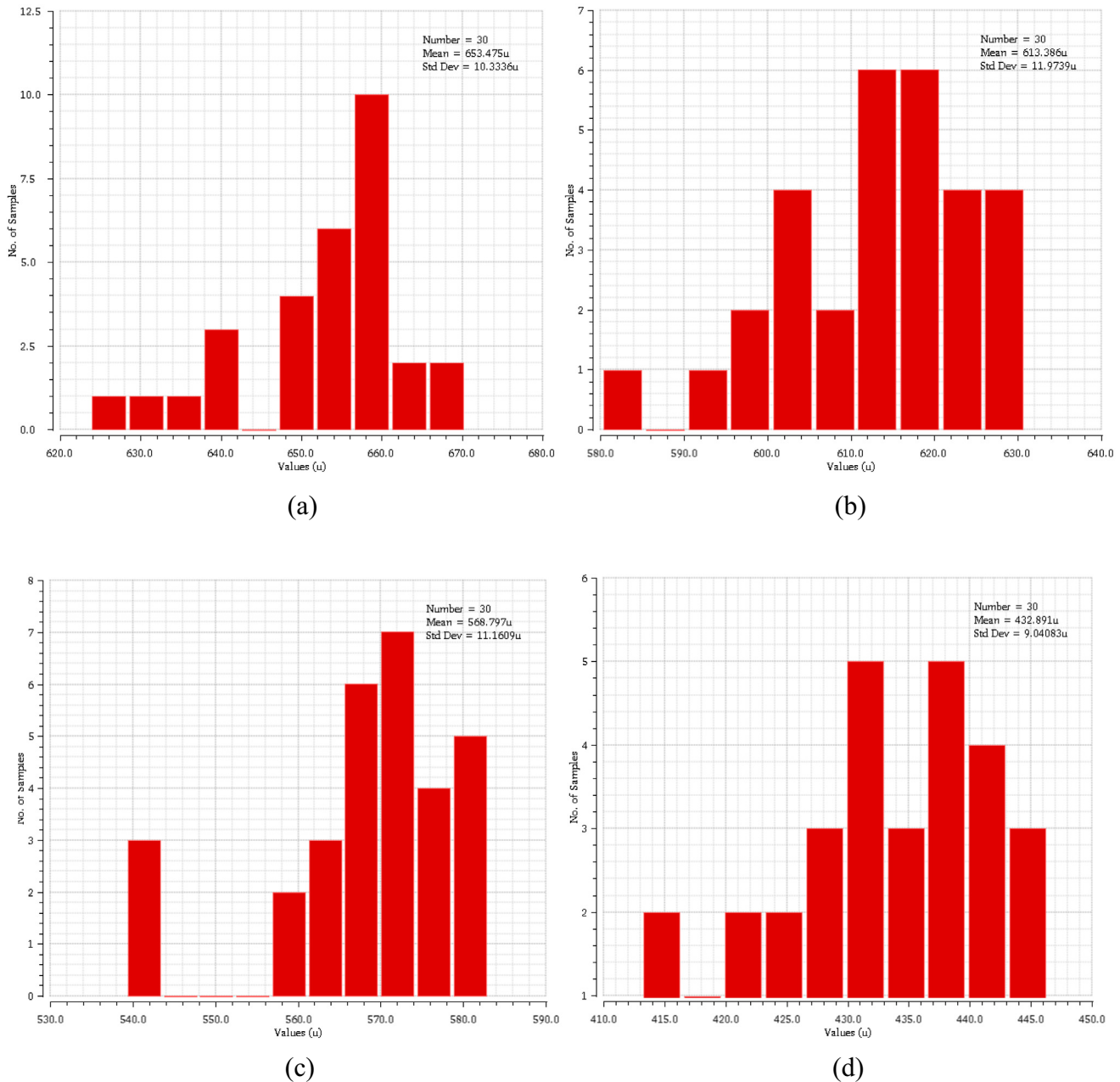


Fig. 6. Monte Carlo simulation results of power distribution of multiplier based on (a) Multiplier [2] (b) Hierarchy [7] (c) Wallace [30] and (c) Proposed hierarchy.

Table 2
Power consumption (μ W) results of 16 bit hierarchy multiplier.

Supply voltage (V)	Multiplier [2]	Hierarchy [7]	Wallace [30]	Proposed multiplier
0.7	235	245	274	212
0.8	354	357	372	306
0.9	472	472	443	356
1.0	578	620	503	414
1.1	658	608	563	424
1.2	716	760	775	481
1.3	868	925	956	556
1.4	1027	1090	1130	644
1.5	1200	1400	1478	756

Table 3
Delay (ps) results of 16 bit hierarchy multiplier.

Supply voltage (V)	Multiplier [2]	Hierarchy [7]	Wallace [30]	Proposed multiplier
0.7	993	903	978	855
0.8	907	803	894	793
0.9	872	723	785	649
1.0	751	661	698	589
1.1	727	594	657	528
1.2	682	509	634	483
1.3	639	477	572	400
1.4	587	446	493	356
1.5	560	397	401	323

Table 4Power Consumption (μW) results of 16 bit hierarchy multiplier.

Load capacitance (fF)	Multiplier [2]	Hierarchy [7]	Wallace [30]	Proposed multiplier
2	658	608	563	424
8	716	775	760	481
14	868	956	925	556
20	1027	1130	1090	644
26	1200	1478	1400	756
32	1389	1613	1569	998

Table 5

Delay (ps) results of 16 bit hierarchy multiplier.

Load capacitance (fF)	Multiplier [2]	Hierarchy [7]	Wallace [30]	Proposed multiplier
2	727	594	657	528
8	782	689	738	563
14	839	777	803	600
20	987	896	951	656
26	1160	997	1108	723
32	1276	1080	1295	817

4.2.4. Area

The layouts are drawn for all the simulated multiplier and the area is calculated from them and listed in Table 6. From the obtained results, it is observed that the proposed multiplier has

30% less area compared with the recently reported Vedic multiplier [15]. This is possible due to replacement of full adder by 4-2 compressor, which in turn minimizes the gates count thus results with small area. The layout of the proposed multiplier is given in Fig. 7.

4.2.5. Sensitive to process variation

A study of circuits performance under the local and global process variations is carried through Monte Carlo simulations with thousand runs ($N = 1000$) and the results are tabulated in Table 7. It is observed that the proposed multiplier has 2% performance variation with respect to process changes. Moreover, the multiplier design based on multichannel technique, discussed in [15] is more sensitive because of driving current dependency on the process sensitivity V_t , which is amplified due to voltage drops at internal nodes.

4.3. 4-2 Compressor

The simulation results of the proposed and existing compressors are given in Table 8. Further, the compressors performance variations in respect of process changes are studied by Monte Carlo simulation.

4.3.1. Delay

Table 8 lists the delay values of the existing and the proposed compressors. It is observed that the proposed compressor has

Table 6

Simulation results of 8 bit multiplier.

S. no.	Design	Delay (ps)	Power consumption (μW)	PDP ($\text{e}-15 \text{ J}$)	Area (μm^2)	Frequency (GHz)
1	Pushpangadan et al. [8]	552	83	45.8	2415	1.811
2	Kayal et al. [15]	465	78	36.2	1678	2.15
3	Proposed Vedic multiplier	432	68	29.3	1164	2.31

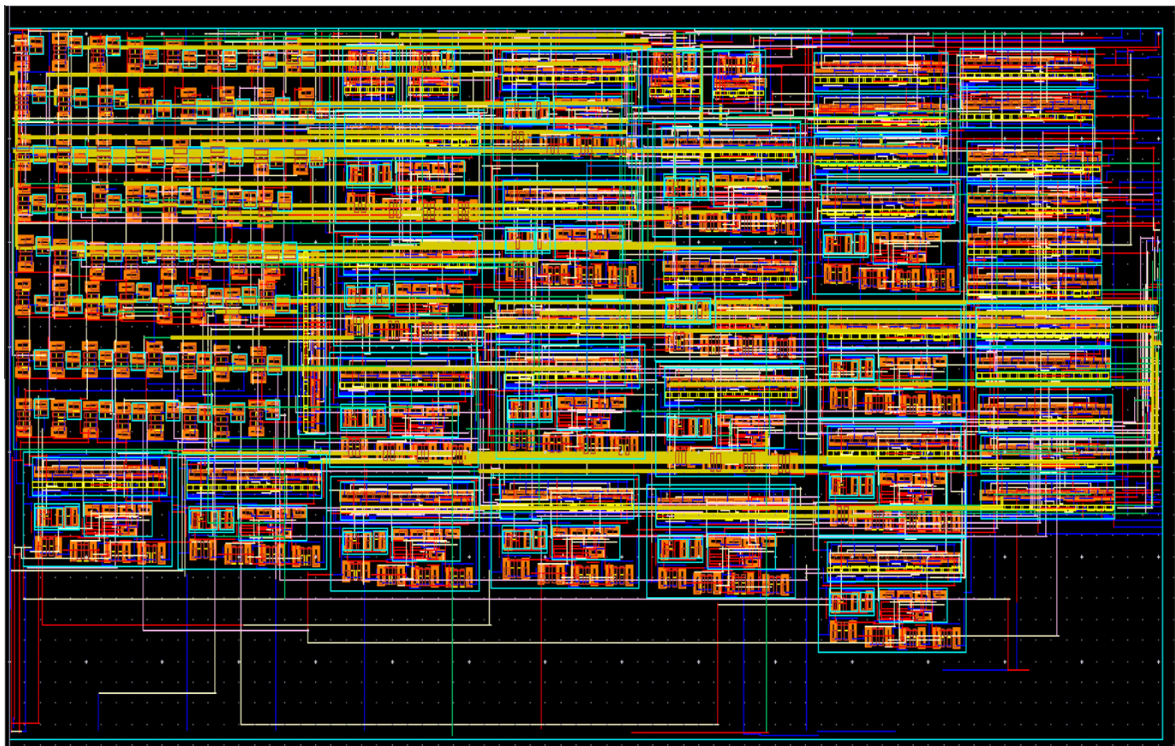
**Fig. 7.** Layout of the proposed 8 bit multiplier.

Table 7
Monte Carlo simulation results of 8 bit multiplier.

S. no.	Design	Delay (ps)	Power consumption (μ W)	PDP ($e-15$ J)
1	Pushpangadan et al. [8]	554	84	46.5
2	Kayal et al. [15]	476	80	38.1
3	Proposed Vedic multiplier	434	69	29.9

Table 8
Simulation results of 4-2 compressor design.

S. no.	Design	Delay (ps)	Power consumption (μ W)	PDP ($e-18$ J)	Area (μm^2)
1	Oklobdzija [18]	175	8.3	1452	55
2	Hussin et al. [19]	137	6.9	945	58
3	Pishvaie et al. [20]	126	6.7	844	56
4	Proposed compressor	114	4.4	502	51

small delay due to parallel computation of intermediate outputs. The speed improvement obtained by the proposed design is 35%, 17% and 10% more than that of compressors discussed by Oklobdzija [18], Hussin et al. [19] and Pishvaie et al. [20], respectively.

4.3.2. Power

As seen from the power consumption results of compressor, given in Table 8, the proposed one can be operated with lesser power consumption than existing designs. This is accomplished with the help of simple architecture and their implementation in FS-GDI logic. Due to sharing of architecture between sum and carry output, the redundant transistor is eliminated thus results in conservation of power.

4.3.3. PDP

It is examined from simulated compressors PDP values given in Table 8, the design discussed by Oklobdzija [18] has more value of

PDP. It is noted that the energy saving accomplished with proposed design is 41% more compared to that of the compressor reported by Pishvaie et al. [20].

4.3.4. Area

The area is calculated from their layout and it is given in Table 8. It is witnessed that proposed compressor requires small area. As stated earlier, the sharing of architecture between sum and carry output reduces the transistor count. Therefore, the layout area of the proposed compressor is lesser and its pictorial form is depicted in Fig. 8. The percentage of area reduction possible with proposed design is about 10% less than that of a recently reported compressor.

4.3.5. Sensitivity to process variation

The operational behavior of the simulated compressors under process changes is studied through Monte Carlo simulations with the thousand runs ($N = 1000$). The circuit's dynamic range of delay and power consumption is analyzed and their respective mean values of the designs are tabulated in Table 9. The complementary pass transistor logic based designs discussed by (Oklobdzija [18], Hussin et al. [19]) are subject to more changes because their performances are more prone to threshold voltage variation. Owing to the introduction of full swing output, the proposed compressor has 1% performance deviation.

4.4. Carry select adder

The simulation results of the existing and proposed CslAs are shown in Table 10.

4.4.1. Delay

As seen from the delay values given in Table 10, the proposed CslA is having 37% reduced delay compared with the recently discussed design in [24]. This is achieved by independent carry computation for both carry in 0 and 1.

4.4.2. Power consumption

The power consumption results reveal that the proposed design operated with less value. This is due to the removal of redundant

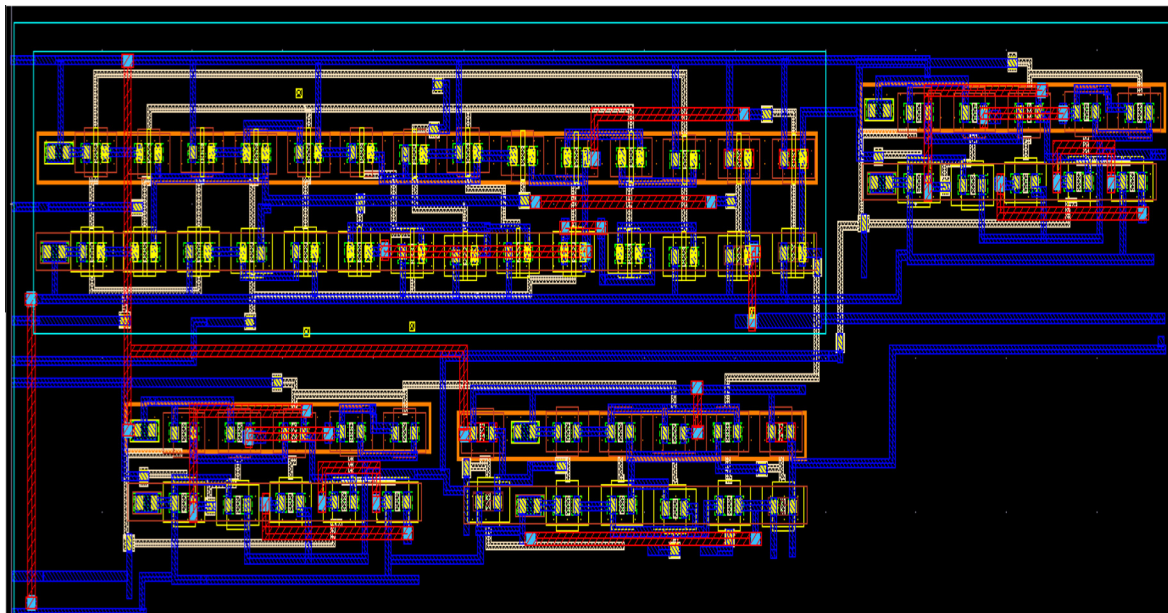


Fig. 8. Layout of the proposed 4-2 compressor.

Table 9

Monte Carlo simulation results of 4-2 compressor.

S. no.	Design	Delay (ps)	Power consumption (nW)	PDP (e–18 J)
1	Oklobdzija [18]	184	8506	1565
2	Hussin et al. [19]	143	6811	0973
3	Pishvaie et al. [20]	129	6855	0884
4	Proposed compressor	115	4410	0507

Table 10

Simulation results of 16 bit CslA adder.

S. no.	Design	Delay (ps)	Power consumption (μW)	PDP (e–15 J)	Area (μm ²)
1	Sqrt CslA [22]	507	129	65	1967
2	BEC CslA [23]	706	93	44	2013
3	Modified CslA [24]	585	85	49	2174
4	Proposed CslA	467	74	35	883

gates presented in the existing design. Not only that, FS-GDI logic minimized the power consumption compared to the CMOS logic based existing design. The power saving in the proposed CslA is 42% and 21% more than the design discussed in [22] and [23], respectively.

4.4.3. PDP

The power delay product values of all the considered CslA adders are given in Table 10. The proposed CslA has 29% lesser value than the recently reported CslA design. Though BEC CslA based design operates with less value among the existing designs, the delay value is more.

4.4.4. Area

The area of CslA adders is calculated from their respective drawn layout. It is examined from the results that proposed one occupies less area. As stated earlier, the proposed one uses partial architecture of carry in 0 for carry in 1 thus minimizes the requirement of hardware, hence, the layout area is small and the layout picture is illustrated in Fig. 9.

Table 11

Simulation results of 8 bit BEC.

S. no.	Design	Delay (ps)	Power consumption (μW)	PDP (e–18 J)	Area (μm ²)
1	CMOS [25]	203	15	3045	537
2	CPL [26]	188	21	3948	583
3	GDI [28]	245	11	2695	501
4	FS-GDI [29]	173	9	1557	445

4.4.5. Sensitive to process variation

The impact of process variation on adder's design is studied from their Monte Carlo simulation. This simulation is carried out for thousand iterations and the CslA adder's performance in respect of delay and power consumption is noted. It is observed that the proposed design has shown 2% performance variation in respect of process changes.

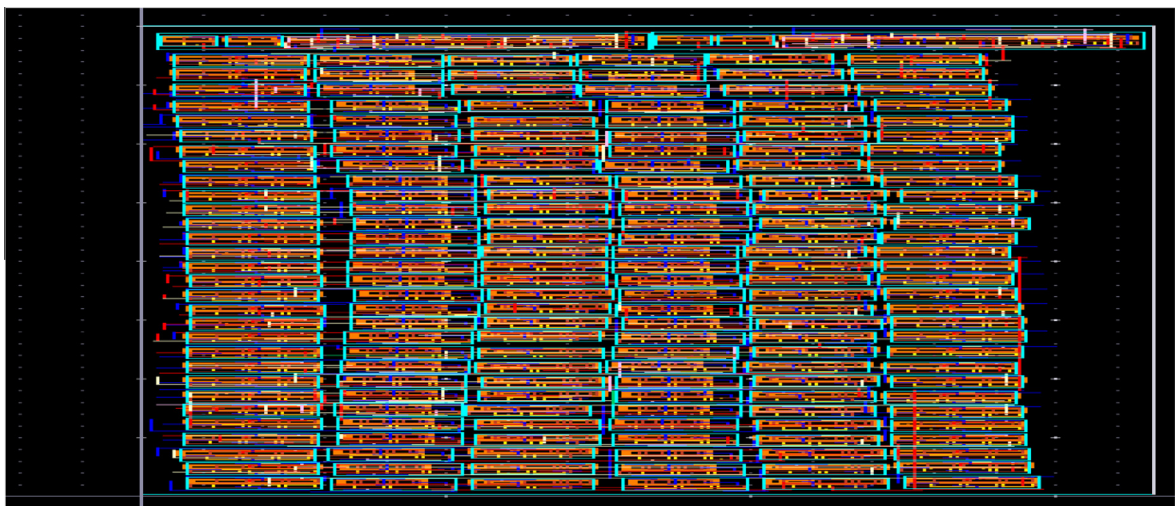
4.5. Binary to Excess 1 Converter

The BEC converter is designed and simulated for CMOS, Complementary Pass Transistor Logic (CPL), GDI and FS-GDI logic. The performance parameters in respect of delay and power consumption are calculated from the simulation results and tabulated in Table 11. As seen from the values, the realization of BEC using FS-GDI logic improves its performance compared with the CMOS and CPL.

The delay and power consumption of the BEC based on FS-GDI logic is reduced by 15% and 40%, respectively compared with conventional CMOS realization [25]. The area is calculated from their layout and given in Table 11. It is observed that the 43% more area saving possible with FS-GDI based BEC design than GDI logic. The layout of BEC using FS-GDI logic is shown in Fig. 10. Further, Monte Carlo simulation is also performed to study the circuit robustness under process variation. From the results, it is noted that BEC circuit based on FS-GDI logic has shown 1% performance variation with respect to process changes.

5. Conclusion

The BEC converter based hierarchy multiplier architecture is proposed which operates with less delay due to the removal of

**Fig. 9.** Layout of 16 bit proposed CslA adder.

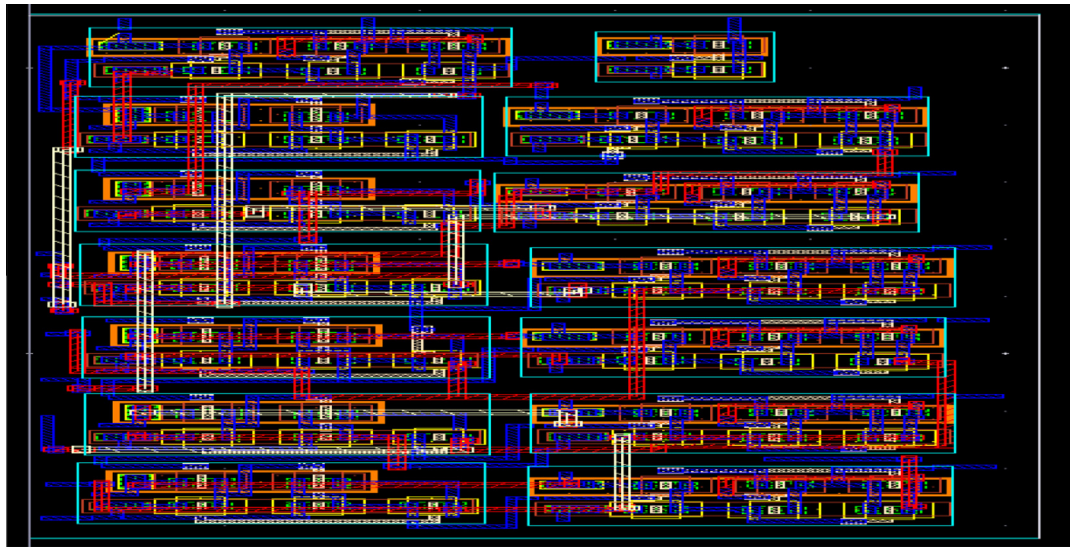


Fig. 10. Layout of 8 bit BEC based on FS-GDI logic.

$n/4$ number of adders, presented in the existing hierarchy multiplier. Moreover, the delay incurred by BEC is not affecting the hierarchical multiplier because it is not included in the critical path of the multiplier. In addition to that, a new design for base multiplier is proposed, based on Vedic mathematics, which is having less delay and area compared with other multipliers found in the literature. The major outcome of the proposed design is the number of adders reduced compared to the other reported works is more. Also, the realization of proposed multiplier using FS-GDI logic reduces the power consumption and area. Thus, area-power and delay efficient hierarchy multiplier is designed. The performances delay and power consumption of the existing and the proposed hierarchy multipliers are calculated through SPICE simulation using 45 nm technology model. From the simulation results, it is calculated that the energy saving achieved by the proposed multiplier design is 17% more than the recently reported multiplier. Further, the multipliers performance study with respect to process variations is done and examined that the proposed multiplier has shown 3% performance variation, which is less than their counterparts. Therefore, the proposed multiplier can be used in the media processing applications in which large width multiplier with less energy consumption is prime importance.

Acknowledgements

This work is supported in part by the University Grants Commission (UGC) India, under the Junior Research Fellowship (JRF) scheme. The authors would like to thank the VIT University, Vellore, India for providing support to carry out some of the simulation works at Integrated Circuit Design Laboratory.

References

- [1] B. Parhami, *Computer Arithmetic: Algorithms and Hardware Designs*, second ed., Oxford University Press, 2010.
- [2] M. Jhamb Garima, H. Lohani, Design, implementation and performance comparison of multiplier topologies in power-delay space, *Eng. Sci. Technol., Int. J.* 19 (2016) 355–363.
- [3] Z. Zakaria, S.A. Abbasi, Optimized multiplier based upon 6 input LUTs and Vedic mathematics, *World Acad. Sci. Eng. Technol.* 7 (2013) 26–30.
- [4] G. Quan, J.P. Davis, S. Devarkal, D.A. Buell, High level synthesis for large bit width multipliers on FPGAs: a case study, in: *Proc. Int. Conf. Hardware/Software Codesign Syst. Synth.*, 2005, pp. 213–218.
- [5] J. Shi, G. Jing, Z. Di, S. Yang, The design and implementation of reconfigurable multiplier with high flexibility, in: *Proceedings of the International Conference on Electronics, Communications and Control*, 2011, pp. 1095–1098.
- [6] S. Quan, Q. Qiang, C.L. Wey, A novel reconfigurable architecture of low power unsigned multiplier for digital signal processing, in: *Proceedings of the International Symposium on Circuits and Systems*, 2005, pp. 3327–3330.
- [7] S.A. Abbasi, Zulhelmi, A.R.M. Alamoud, FPGA design, simulation and prototyping of 32 bit pipeline multiplier based on Vedic mathematics, *IEICE Electron. Exp.* 12 (2015) 1–12.
- [8] R. Pushpangadan, V. Sukumaran, R. Innocent, High speed Vedic multiplier for digital signal processors, *IETE J. Res.* 55 (2009) 282–286.
- [9] A. Ronisha Prakash, S. Kirubaveni, Performance evaluation of FFT processor using conventional and Vedic algorithm, in: *Proceedings of the International Conference Emerging Trends in Computing, Communication and Nanotechnology*, 2013, pp. 89–94.
- [10] K. Sethi, R. Panda, Multiplier less high speed squaring circuit for binary numbers, *Int. J. Electron.* 102 (2015) 433–443.
- [11] M. Ramalatha, K. Thanushkodi, A novel time and energy efficient cubing circuit using Vedic mathematics for finite field arithmetic, in: *Proceedings of the International Conference on Advances in Recent Technologies in Communication and Computing*, 2009, pp. 873–875.
- [12] R. Senapati, B.K. Bhoi, Urdhava triyakhyam sutra: application of Vedic mathematics for a high speed multiplier, *Int. J. Creative Math. Sci. Technol.* 1 (2012) 59–66.
- [13] P. Saha, K. Banerjee, A. Dandapat, P. Bhattacharya, Vedic mathematics based 32 bit multiplier design for high speed low power processors, *Int. J. Smart Sens. Intell. Syst.* 4 (2011) 268–284.
- [14] P. Saha, K. Banerjee, A. Dandapat, P. Bhattacharya, ASIC design of high speed low power circuit for factorial calculation using ancient Vedic mathematics, *Microelectron. J.* 42 (2011) 1343–1352.
- [15] D. Kayal, P. Mostafa, A. Dandapat, C.K. Sarkar, Design of high performance 8 bit multiplier using algorithm with McCMOS technique, *J. Signal Process. Syst.* 76 (2014) 1–9.
- [16] J.S.S.B.K.T. Maharaja, *Vedic Mathematics*, first ed., Motilal Banarsidass Press, 2001.
- [17] N. Nagamatsu, S. Tanaka, J. Mori, T. Noguchi, H. Hatanaka, A 15 ns 32×32 -bit CMOS multiplier with an improved parallel structure, *IEEE J. Solid-State Circuits* 25 (1990) 494–497.
- [18] V.J. Oklobdzija, Improving multiplier design by using improved column tree and optimized final adder in CMOS technology, *IEEE Trans. Very Large Scale Integr. VLSI Syst.* 3 (1995) 292–301.
- [19] R. Hussin, A.Y.M. Shakaff, N.S.Z. Idris, R.C. Ismail, A. Kamarudin, An efficient modified booth multiplier architecture, in: *Proceedings of the International Conference on Electronic Design*, 2008, pp. 1–4.
- [20] A. Pishvaie, G. Jaberipur, A. Jahanian, Redesign CMOS 4:2 compressor for fast binary multipliers, *Can. J. Electr. Comput. Eng.* 36 (2013) 111–115.
- [21] A. Fathi, S. Azizian, K. Hadidi, A. Khoei, A. Chegani, CMOS implementation of a fast 4-2 compressor for parallel accumulations, in: *Proceedings of the International Symposium on Circuits and Systems*, 2012, pp. 1476–1479.
- [22] M.A. Chandrakasan, R.W. Broderson, *Low Power Digital CMOS Design*, fourth ed., Kluwer Academic Publishers, 2003.
- [23] B. Ramkumar, H.M. Kittur, Low-power and area-efficient C_{SLA}, *IEEE Trans. Very Large Scale Integr. Syst.* 20 (2012) 371–375.
- [24] B.K. Mohanty, S.K. Patel, Area-delay-power efficient carry-select adder, *IEEE Trans. Circuits Syst. I Reg. Pap.* 61 (2014) 418–422.
- [25] N.H.E. Weste, D. Harris, *CMOS VLSI Design*, second ed., Pearson Education, 2005.
- [26] S. Purohit, M. Margala, Investigating the impact of logic and circuit implementation for full adder performance, *IEEE Trans. Very Large Scale Integr. VLSI Syst.* 20 (2012) 1327–1331.

- [27] V. Foroutan, M. Teheri, K. Navi, A. Mazreah, Design of two low power full adder using GDI structure and hybrid CMOS logic style, *Integration, VLSI J.* 47 (2014) 48–61.
- [28] A. Morgenshtein, I. Shwartz, A. Fish, Full swing gate diffusion input (GDI) logic – case study for low power CLA adder design, *Integration, VLSI J.* 47 (2014) 62–70.
- [29] M. Shoba, R. Nakkeeran, GDI based full adders for energy efficient arithmetic applications, *Eng. Sci. Technol., Int. J.* 19 (2016) 485–496.
- [30] S. Abed, B. Jamil Mohd, Z. Al- Bayati, S. Alouneh, A low power Wallace multipliers based on wide counters, *Int. J. Circuit Theory Appl.* 40 (2012) 1175–1185.